122589

MET O 11 TECHNICAL NOTE NO:71

METEOROLOGICAL OFFICE
9 SEP 1976
LIBRARY

A DIRECT METHOD FOR THE SOLUTION OF

HELMHOLTZ-TYPE EQUATIONS

BY

M C TAPP

CONTENTS

§1. INTRODUCTION 1

§2. NOTATION AND NOMENCLATURE 1

§3. GENERAL THEORY: THE LYAPUNOV EQUATION 2

§4. APPLICATIONS TO SPECIAL CASES 5

§5. DECOUPLING/COUPLING PROCEDURE: USE OF FAST FOURIER TRANSFORM METHODS 9

§6. COMPARISON WITH ITERATIVE METHODS 14

§1. INTRODUCTION

Nearly always at some stage in the numerical analysis and prediction of meteorological fields, a 2-D (perhaps even 3-D) second order partial differential equation has to be solved for some particular scalar quantity. This note describes a useful direct method for solving a certain class of equations which, typically, crop up in semi-implicit prediction models. In particular this note came about from the need to find a method for solving the equation

$$-\nabla_H^2\phi + \lambda\phi = F, \qquad (1)$$

for small λ. Here ∇_H^2 is the horizontal Laplacian, $\lambda > 0$ is a constant and $F = F(x,y)$ is a field defined over a rectangular domain. For small values of λ in (1) iterative methods fail because of the difficulty of obtaining a 'reasonable' first guess solution.

Direct methods are often dismissed on the grounds that they demand too much arithmetic. This note shows that, far from being time consuming, direct methods can be made very competitive with iterative methods of solution.

§2 is devoted to the notation and nomenclature used throughout this note. The Lyapunov equation is discussed in §3 together with the necessary and sufficient conditions for a unique solution and also direct methods of solution available. In §4 some applications to special cases are given with particular emphasis on Neumann and Dirichlet boundary conditions. §5 deals with the use of fast Fourier transform (FFT) methods in computing the decoupling/coupling part of the solution given in §4. In §6 a comparison is made between the method developed here and the efficient alternating direction implicit (ADI) iterative method.

§2. NOTATION AND NOMENCLATURE

Since this note depends to a large extent on the manipulation and use of matrices, a short description of the notation and conventions used is given here. A matrix B is denoted by $\underset{\approx}{B}$ and a particular element of $\underset{\approx}{B}$ by B_{ij} where the j suffix refers to rows and the i suffix to columns. $K_i(\underset{\approx}{B})$ denotes the i th column of $\underset{\approx}{B}$. If the matrix has n rows and m columns it will be called a matrix of order n x m. The matrix $\underset{\approx}{\Lambda}$ is used to denote a square matrix whose elements are zero apart from

those along the main diagonal. $\underset{\approx}{I}$ is the unity matrix defined by

$$\underset{\approx}{I}\,\underset{\approx}{B} = \underset{\approx}{B}\,\underset{\approx}{I} = \underset{\approx}{B}$$

for any $\underset{\approx}{B}$.

In §5 Fourier transform methods are discussed. This is a one-to-one mapping of any sequence A_n of complex numbers, n=o, 1,...m-1 onto another sequence X_j, j=0,1,...m-1, defined by

$$X_j = \sum_{n=0}^{m-1} A_n W_m^{nj}$$

where W_m^{nj} is understood to mean $(W_m)^{nj}$ and

$$W_m = EXP[2\pi i/m] \quad , \quad i = \sqrt{-1},$$

is the principal mth root of unity. The above transformation is written for brevity as

$$X_j \longleftrightarrow A_n$$

§3. GENERAL THEORY: THE LYAPUNOV EQUATION

The Lyapunov equation is best known in its form

$$\underset{\approx}{Y}\,\underset{\approx}{\Phi} + \underset{\approx}{\Phi}\,\underset{\approx}{X} = \underset{\approx}{C} \quad , \tag{2}$$

where the $\underset{\approx}{C}$ and $\underset{\approx}{\Phi}$ matrices are of order n x m, $\underset{\approx}{X}$ is of order m x m and $\underset{\approx}{Y}$ is of order n x n. Equation (2) plays a fundamental role in the equilibrium or stability theory of differential systems (Bellman 1969), a topic started independently by Poincare and Lyapunov. It can be shown (Bellman 1970) that the necessary and sufficient condition for (2) to have a unique solution in $\underset{\approx}{\Phi}$ for all $\underset{\approx}{C}$ is that $\lambda_j + \mu_i \neq 0$, where λ_j are the eigenvalues of $\underset{\approx}{Y}$ and μ_i are the eigenvalues of $\underset{\approx}{X}$.

The Lyapunov equation, in connection with the solution of differential systems, has been the subject of some study in the literature (eg Bickley and McNamee 1960, Osborne 1965). Bickley and McNamee give three methods for solving equation (2), which they term the semi-rational, irrational and rational methods, to be described below.

SEMI-RATIONAL METHOD

This appears the most useful of the three and is the one which is developed further in this note. The method requires a knowledge of the similarity transform of either $\underset{\approx}{X}$ or $\underset{\approx}{Y}$. If that of $\underset{\approx}{X}$ is known then it is possible to write

$$\underset{\approx}{X} = \underset{\approx}{E}\,\underset{\approx}{\Lambda}_x\,\underset{\approx}{E}^{-1},$$

where the non-zero elements of $\underset{\approx}{\Lambda}_x$ are the eigenvalues of $\underset{\approx}{X}$. Upon substitution for $\underset{\approx}{X}$ in (2) and defining

$$\underset{\approx}{\zeta} = \underset{\approx}{\Phi}\,\underset{\approx}{E} \quad \& \quad \underset{\approx}{\bar{C}} = \underset{\approx}{C}\,\underset{\approx}{E},$$

then

$$\underset{\approx}{Y}\,\underset{\approx}{\zeta} + \underset{\approx}{\zeta}\,\underset{\approx}{\Lambda}_x = \underset{\approx}{\bar{C}}. \qquad (3)$$

Equation (3) can now easily be broken down into sets of simultaneous linear equations of the form

$$\left(\underset{\approx}{Y} + \mu_i \underset{\approx}{I}\right) K_i\left(\underset{\approx}{\zeta}\right) = K_i\left(\underset{\approx}{\bar{C}}\right), \quad i = 1, 2, \ldots m. \qquad (4)$$

Each column of $\underset{\approx}{\zeta}$ can be solved from (4) and the solution completed by a matrix multiplication. In the case of matrices of large order the matrix multiplications in calculating $\underset{\approx}{\bar{C}}$ and $\underset{\approx}{\Phi}$ can be a large, if not the dominating part of the computation. This will be returned to in §5. Equation (4) only applies when the eigenvalues of $\underset{\approx}{X}$ are all distinct and that is the only case considered here (but see Osbourne (1965) when this is not so).

IRRATIONAL METHOD

The irrational method involves a knowledge of the similarity transforms of $\underset{\approx}{Y}$ and $\underset{\approx}{X}$. For $\underset{\approx}{Y}$ it is assumed that

$$\underset{\approx}{Y} = \underset{\approx}{S}\,\underset{\approx}{\Lambda}_Y\,\underset{\approx}{S}^{-1},$$

where $\underset{\approx}{\Lambda}_Y$ is diagonal with elements λ_j. Then

$$\left(\underset{\approx}{\Lambda}_Y + \mu_i \underset{\approx}{I}\right) K_i\left(\underset{\approx}{S}^{-1}\underset{\approx}{\zeta}\right) = K_i\left(\underset{\approx}{S}^{-1}\underset{\approx}{\bar{C}}\right). \qquad (5)$$

Since $\underset{\approx}{\Lambda}_Y$ is diagonal the solution of (5) is trivial. Then the solution to (2) may be found by two matrix multiplications. Since the elements of $\underset{\approx}{\Lambda}_Y$ are the λ_j's, (5) also demonstrates the uniqueness condition given earlier, since

$$(\lambda_j + \mu_i)\left(\underset{\approx}{S}^{-1}\underset{\approx}{\zeta}\right)_{ij} = \left(\underset{\approx}{S}^{-1}\underset{\approx}{\bar{C}}\right)_{ij},$$

for each i and j. Providing $\lambda_j + \mu_i \neq 0$ a solution for $\underset{\approx}{\Phi}$ can always

be found. Hence at least one of the two matrices $\underset{\approx}{X}$ and $\underset{\approx}{Y}$ must be non-singular.

RATIONAL METHOD

The rational method of solution requires a knowledge of the characteristic equation of either $\underset{\approx}{X}$ or $\underset{\approx}{Y}$. The characteristic equation ψ of a matrix $\underset{\approx}{A}$ is defined by evaluating the determinant $|\underset{\approx}{A}-\lambda\underset{\approx}{I}|$; ie

$$\psi(\lambda) = |\underset{\approx}{A}-\lambda\underset{\approx}{I}| = \sum_{r=0}^{n-1} a_r\lambda^r + \lambda^n ,$$

and is a polynomial of degree n in λ if $\underset{\approx}{A}$ is of order n x n. The Cayley-Hamilton theorem (see for example Bellman 1970) states that the matrix $\underset{\approx}{A}$ satisfies its own characteristic equation; ie

$$\psi(\underset{\approx}{A}) = \underset{\approx}{O}.$$

$\underset{\approx}{A}^r$ is understood to mean the result of the r-1 matrix multiplications $\underset{\approx}{A}\underset{\approx}{A}\ldots\underset{\approx}{A}$ (r terms). In particular if that of $\underset{\approx}{Y}$ is known then $\psi(\underset{\approx}{Y}) = \underset{\approx}{O}$ or

$$\sum_{r=0}^{n-1} a_r\underset{\approx}{A}^r + \underset{\approx}{A}^n = \underset{\approx}{O} ,$$

where $\underset{\approx}{O}$ is the n x n null matrix (all elements zero). Clearly since $\psi(\underset{\approx}{Y}) = \underset{\approx}{O}$, then

$$\psi(\underset{\approx}{Y})\underset{\approx}{\Phi} = \underset{\approx}{O} .$$

From equation (2) it is fairly straightforward to establish the identity

$$\underset{\approx}{Y}^r\underset{\approx}{\Phi} = \underset{\approx}{\Phi}(-\underset{\approx}{X})^r + \sum_{i=0}^{r-1}(-1)^i\underset{\approx}{Y}^{r-i-1}\underset{\approx}{C}\underset{\approx}{X}^i$$

and hence the equation

$$\underset{\approx}{O} = \psi(\underset{\approx}{Y})\underset{\approx}{\Phi} = \underset{\approx}{\Phi}\psi(-\underset{\approx}{X}) + \underset{\approx}{R}(\underset{\approx}{Y}, \underset{\approx}{C}, \underset{\approx}{X}). \qquad (6)$$

Equation (6) is a set of linear equations in the rows of $\underset{\approx}{\Phi}$. The matrix $\underset{\approx}{R}$ represents those terms independent of $\underset{\approx}{\Phi}$.

In application the first method (semi-rational) is to be preferred providing the resulting sets of linear equations can be handled efficiently. This is the method we develop further in §4 when some important special cases of equation (2) are discussed. The other two methods of solution proposed by Bickley and McNamee do not appear to offer any advantages over the first method, especially when considering matrix equations of large order.

§ 4. APPLICATIONS TO SPECIAL CASES

In general, of course, the application of the semi-rational method to large systems would involve a considerable amount of computation in both the matrix multiplications and the solution of the sets of simultaneous equations. The matrix multiplications needed to compute $\underset{\approx}{\bar{C}}$ and $\underset{\approx}{\bar{\Phi}}$ each involve $n \times n^2$ multiplications and $nm(m-1)$ additions. To solve the n sets of equations (4) directly by, say, Gaussian elimination with complete pivoting involves approximately $m \times n^3/3$ multiplications and about the same number of additions. So that on a rectangular mesh n x m, the number of operations (both additions and multiplications) per grid point is roughly $4m + 2n^2/3$. Indeed in the general case the irrational method would be more favourable since this involves four matrix operations resulting in about $4m + 4n$ multiplications and additions per grid point. However, for large systems, problems may be encountered in computing the similarity transforms and their inverses.

The above computation estimates for the general case indicate that direct methods must be applied cautiously and to special systems where it is possible to drastically reduce the number of floating point operations. In this respect attention is now turned to the equation presented in the introduction to this note, namely the Helmholtz equation

$$-\nabla_H^2 \phi + \lambda \phi = F \quad . \tag{7}$$

It is assumed that F is a 2-D field defined over a uniform retangular mesh of points (n x m). The standard finite difference form of (7) can be written as

$$-\phi_{i+1,j} - \phi_{i-1,j} - \phi_{i,j+1} - \phi_{i,j-1} + (\lambda + 4)\phi_{ij} = F_{ij} . \tag{8}$$

(8) may be rewritten formally as

$$[X]\phi_{ij} + [Y]\phi_{ij} + \lambda \phi_{ij} = F_{ij} \tag{9}$$

where $[X]$ and $[Y]$ are formal operators such that

$$[Y]\phi_{ij} = -\phi_{i,j+1} - \phi_{i,j-1} + 2\phi_{ij}$$

and

$$[X]\phi_{ij} = -\phi_{i+1,j} - \phi_{i-1,j} + 2\phi_{ij}$$

If (9) is transformed into matrix notation with

$$\underset{\approx}{F} = \{ F_{ij} \} \quad ,$$

and

$$\underset{\approx}{\Phi} = \{ \phi_{ij} \} \quad ,$$

then the $[X]$ operator must be a tri-diagonal matrix ($\underset{\approx}{X}$) which multiplies $\underset{\approx}{\Phi}$ on the right. Similarly the $[Y]$ operator is tri-diagonal ($\underset{\approx}{Y}$) and multiplies $\underset{\approx}{\Phi}$ on the left. (9) can then be written as

$$\lambda \underset{\approx}{\Phi} + \underset{\approx}{Y}\underset{\approx}{\Phi} + \underset{\approx}{\Phi}\underset{\approx}{X} = \underset{\approx}{F} \quad . \tag{10}$$

If $\underset{\approx}{\Phi}$ is n x m, then $\underset{\approx}{X}$ will be m x m and $\underset{\approx}{Y}$ n x n. Thus Helmholtz equations of the form (7) can be reformulated in terms of a Lyapunov equation (10). The non-zero elements of $\underset{\approx}{X}$ or $\underset{\approx}{Y}$ are -1, 2, -1 except for the first and last rows which must be modified to include boundary conditions.

1) NEUMANN BOUNDARY CONDITIONS

In this case the value of $\frac{\partial \phi}{\partial n}$ around the boundary is known. The matrix operators $\underset{\approx}{X}$ or $\underset{\approx}{Y}$ now take the form

$$\begin{bmatrix} 1 & -1 & 0 & \cdots & & 0 \\ -1 & 2 & -1 & 0 & \cdots & 0 \\ \vdots & & \ddots & & & \vdots \\ & & & \ddots & & \\ 0 & & & \cdots & -1 & 2 & -1 \\ 0 & & & \cdots & 0 & -1 & 1 \end{bmatrix} \tag{11}$$

Employing the semi-rational method of solution to equation (10) requires computing the similarity transform for $\underset{\approx}{X}$. If $\underset{\approx}{X}$ has the form (11) it is fairly easy to show that the eigenvalues are

$$\mu_i = 4 \sin^2 \frac{(i-1)\pi}{2m} , \quad i = 1, 2, \ldots m$$

and the normalised (to one) eigenvectors

$$\underline{x}_j = \sqrt{\frac{2}{m}} \left\{ \cos \left[\frac{(j-\tfrac{1}{2})(i-1)\pi}{m} \right] \right\} ,$$

where the elements of $\underline{x}_j$ are given by i=1, 2, ...m, and the different eigenvectors are j=1, 2, ...m.

By forming the matrix $\underset{\approx}{E}$ whose columns are the eigenvectors of $\underset{\approx}{X}$, then

$$\underset{\approx}{X} = \underset{\approx}{E}\,\underset{\approx}{\Lambda}_x\,\underset{\approx}{E}^{-1}, \quad \underset{\approx}{\Lambda}_x = \underset{\approx}{DIAG}(\mu_i).$$

Furthermore since the eigenvalues of $\underset{\approx}{X}$ are distinct, it can be shown that the eigenvectors form an orthogonal set of dimension m, so that

$$\underset{\approx}{E}^{-1} = \underset{\approx}{E}^{T} \quad \text{(transpose)}.$$

Substituting for $\underset{\approx}{X}$ in (10) yields

$$\lambda \underset{\approx}{\Phi} + \underset{\approx}{Y}\,\underset{\approx}{\Phi} + \underset{\approx}{\Phi}\,\underset{\approx}{E}\,\underset{\approx}{\Lambda}_x\,\underset{\approx}{E}^{-1} = \underset{\approx}{F} \qquad (12)$$

or

$$\left(\underset{\approx}{Y} + (\lambda+\mu_i)\underset{\approx}{I}\right) K_i(\underset{\approx}{\zeta}) = K_i(\underset{\approx}{\bar{F}}), \quad i=1,\ldots m, \qquad (13)$$

where

$$\underset{\approx}{\zeta} = \underset{\approx}{\Phi}\,\underset{\approx}{E} \quad \text{and} \quad \underset{\approx}{\bar{F}} = \underset{\approx}{F}\,\underset{\approx}{E}$$

Since $\underset{\approx}{Y}$ is tri-diagonal and $\lambda > 0$, $\mu_i \geq 0$, $\left(\underset{\approx}{Y} + (\lambda+\mu_i)\underset{\approx}{I}\right)$ is diagonally dominant and the standard tri-diagonal algorithm given by Varga (1962, p.195) may be used (see also Richtmyer and Morton 1967). This requires approximately four additions, two multiplications and one division per point and is independent of the number of points. The method, which is a special adaption of Gaussian elimination with pivoting about each column, is often referred to in the literature as the double sweep method. A sweep is made through the data in the order of increasing j determining two sets of coefficients inductively for j=1, 2, ...n-1. The boundary condition for j=1 is included in this calculation and that for j=n essentially gives the solution for that point. To develop the rest of the solution merely requires a sweep through the data in order of decreasing j. The method is very efficient and keeps all the coefficients nicely in scale.

Using the double-sweep method in the semi-rational solution means that the matrix multiplications are the only major part of the computation. Before going on to discuss possible ways of reducing this part of the calculation, one further special case will be discussed.

ii) DIRICHLET BOUNDARY CONDITIONS

In this case the value of ϕ is given around the boundary. The matrices $\underset{\approx}{X}$ or $\underset{\approx}{Y}$ now take the form

$$\begin{bmatrix} 2 & -1 & 0 & \cdots & & & 0 \\ -1 & 2 & -1 & 0 & \cdots & & 0 \\ \vdots & \ddots & \ddots & \ddots & & & \vdots \\ & & & & & & \\ \vdots & & & & & & \vdots \\ 0 & & & \cdots & -1 & 2 & -1 \\ 0 & & & \cdots & 0 & -1 & 1 \end{bmatrix} \qquad (14)$$

The matrix $\underset{\approx}{Y}$ can now be shown to possess the eigenvalues

$$\mu_i = 4 \sin^2 \frac{i\pi}{2(m+1)}, \quad i = 1, 2, \ldots m,$$

and the normalised eigenvectors

$$\underset{\sim}{x}_j = \sqrt{\frac{2}{m+1}} \left\{ \sin \frac{ji\pi}{(m+1)} \right\}, \quad j, i = 1, 2, \ldots m.$$

Unlike the Neumann case there is no zero eigenvalue. Again equation (13) holds with $\underset{\approx}{Y}$ given by (14) and $\lambda > 0$, $\mu_i > 0$ so that the matrix $(\underset{\approx}{Y} + (\lambda + \mu_i)\underset{\approx}{I})$ is still diagonally dominant and the tri-diagonal algorithm can be applied. The matrix $\underset{\approx}{F}$ (formed from the eigenvectors of $\underset{\approx}{X}$) in this case is also its own inverse.

In conclusion it is to be noted that (7) is not the most general system that can be reduced to solving tri-diagonal equations. The most general elliptic equation capable of solution by this method is

$$-\nabla_H^2 \phi + \rho(y)\phi + \beta(y)\frac{\partial \phi}{\partial y} = F, \qquad (15)$$

where β and ρ are functions of y but not of x. The methods already described can be easily adapted to deal with (15).

§5. DECOUPLING / COUPLING PROCEDURE: USE OF FAST FOURIER TRANSFORM METHODS

§4 described how to reduce the Helmholtz equation (7) to solving sets of tri-diagonal equations, and described an efficient method for doing so. However, in decoupling (7) into the normal modes of the $\underset{\approx}{X}$ operator and then re-coupling requires two matrix multiplications namely:

$$\underset{\approx}{\overline{F}} = \underset{\approx}{F}\,\underset{\approx}{E}$$

and

$$\underset{\approx}{\Phi} = \underset{\approx}{\zeta}\,\underset{\approx}{E}^{-1} = \underset{\approx}{\zeta}\,\underset{\approx}{E}^{T}.$$

Each matrix operation requires m multiplications and m-1 additions per point.

The elements of the matrix $\underset{\approx}{E}$ are either sines or cosines depending on the boundary conditions employed. In the case of Dirichlet conditions the matrix multiplications reduce to

$$\overline{F}_{ij} = \sqrt{\frac{2}{m+1}} \sum_{r=1}^{m} F_{rj} \sin\frac{ri\pi}{(m+1)}, \quad j,i = 1,2 \ldots m$$

and

$$\Phi_{ij} = \sqrt{\frac{2}{m+1}} \sum_{r=1}^{m} \zeta_{rj} \sin\frac{ri\pi}{(m+1)}, \quad j,i = 1,2 \ldots m \qquad (15)$$

which are the Fourier sine transforms of F_{rj} and ζ_{rj}. Hence the matrix multiplications represent simply the Fourier decomposition by rows of the data. The fast Fourier transform(FFT) method is a computational algorithm (Cooley et al 1967) which greatly increases the speed at which transforms like (16) can be computed - typically by one order of magnitude.

The discrete complex Fourier transform is a one-to-one mapping between the complex sequences A_n, $n=0,1. \ldots m-1$ and X_j, $j=0,1. \ldots m-1$ as defined in §2 ie

$$X_j \longleftrightarrow A_n$$

Real sine or cosine transformations, like (16), can be manipulated into complex transforms on half the number of points as described by Cooley (1970). Thus in all calculations it is assumed that m is even.

In the case of Neumann boundary conditions the matrix multiplications reduce to

$$\overline{F}_{ij} = \sqrt{\frac{2}{m}} \sum_{r=1}^{m} F_{rj} \cos\left[\frac{(2r-1)(i-1)\pi}{2m}\right] ,$$

$$\Phi_{ij} = \sqrt{\frac{2}{m}} \sum_{r=1}^{m} \zeta_{rj} \cos\left[\frac{(2i-1)(r-1)\pi}{2m}\right], \quad j,i = 1,2,\ldots m \qquad (17a,b)$$

which unfortunately do not appear to be related directly to any Fourier transform. Since Neumann conditions are by their nature more complicated than the Dirichlet type, it is to be expected that transform methods will be more difficult to apply.

There are two distinct transforms to consider in the Neumann case (17a,b); equation (17a) referring to the decoupling into normal modes, equation (17b) referring to the coupling into the final solution. The purpose here it to show how a subroutine or special hardware unit capable of computing the discrete Fourier series.

$$X_j = \sum_{\Lambda=0}^{M-1} A_\Lambda W_m^{\Lambda j} , \quad j = 0, 1, \ldots M-1, \qquad (18)$$

can, with some suitable pre-and post-processing of the data compute the special transforms (17a,b). The reader is referred to Cooley (1970) for detailed information concerning the computation of real Fourier series.

Before proceeding further, some definitions and remarks are made. Firstly, to be consistent with notation often found in the literature, all summations start at zero, so that the transform pair (17a,b) are written as

$$\overline{F}_{i+1,j+1} = \sqrt{\frac{2}{m}} \sum_{r=0}^{m-1} F_{r+1,j+1} \cos\left[\frac{(2r+1)i\pi}{2m}\right] ,$$

$$\Phi_{i+1,j+1} = \sqrt{\frac{2}{m}} \sum_{r=0}^{m-1} \zeta_{r+1,j+1} \cos\left[\frac{(2i+1)r\pi}{2m}\right], \quad j,i = 0,1,\ldots m-1 \qquad (19a,b)$$

An important property of the Fourier transform (18) is that the indices j and n are to be interpreted modulo m. Therefore, X_{-j} with $0 \leq j \leq m-1$ is understood to mean X_{m-j}. Further, a sequence is defined to be even if

$$X_j = X_{m-j},$$

and odd if

$$X_j = -X_{m-j}.$$

Many special properties of X_j translate into special properties of A_n. For example if X_j is real and odd, A_n is imaginery and odd. A list of various properties is given in Cooley (1970).

NEUMANN DECOUPLING PROCEDURE

The purpose here is to develop a procedure for calculating (19a). To make the transform recognisable, the 2m+1 data values $A_r, r = 0,1,\ldots 2m$ are defined as

$$A_r = \{0, F_{1j}, 0, F_{2j}, 0 \ldots 0, F_{mj}, 0\},$$

where A_r (r even) is zero. Then consider the transform

$$Y_i = \sum_{r=0}^{2m} A_r \cos\left[\frac{r i \pi}{2m}\right], \quad i = 0,1,\ldots 2m. \qquad (20)$$

(20) represents a cosine transform on 2m+1 points, the first m-values of the Y_i's being the $\overline{F}_{i+1,j+1}$'s required in (19a). Because of the nature of the A_r's, the problem of computing (19a) has been neatly embedded in that of (20) - and (20) is

a transform which is familiar. That really solves the problem, since one could go ahead and compute all the Y_i's using for example procedure 6 described in Cooley (1970) for a cosine transform. However (20) involves computing (2m+1) values when only m are in fact required. A suitable modification of Cooley's procedure 6 neatly eliminates half the calculation as below.

Equation (20) may be written in terms of an even sequence C_r as

$$Y_i = \sum_{r=0}^{4M-1} C_r W_{4M}^{ri}, \qquad (21)$$

where for odd r, $A_r = 2C_r$, while for even r, $A_r = C_r = 0$. Since C_r is an even sequence, $C_r = C_{4M-r}$. The reader will easily verify that (21) reduces to (20) in this case. Now the manipulations set out by Cooley (1970) require forming the 2m complex numbers

$$X_r = C_{2r} + i(C_{2r+1} - C_{2r-1}), \quad r = 0, 1, \ldots 2M-1,$$

and then computing the transform

$$Z_i = \sum_{r=0}^{2M-1} X_r W_{2M}^{ri}, \quad i = 0, 1, \ldots 2M-1 \qquad (22)$$

Suitable formulae set out in Cooley (1970) give the Y_i in terms of the Z_i. However A_r for even r, is zero, which implies that C_{2r} is zero also. Therefore

$$X_r = i(C_{2r+1} - C_{2r-1}), \quad r = 0, 1, \ldots 2M-1$$

and can easily be shown to be a pure imaginery and odd sequence. This implies that the Z_i's are real and odd, ie $Z_i = -Z_{2M-i}$. Using this fact it is easy to show that (22) becomes simply a sine transform of m-1 points, namely

$$z_i = \sum_{r=1}^{m-1} b_r \sin\left(\frac{r i \pi}{m}\right),$$

where

$$b_r = 2 i X_r, \quad (i = \sqrt{-1}).$$

Hence once the z_i are computed from (23) the appropriate formulae may be used to obtain the Y_i in (21) or (20). Thus the amount of work required is one sine transform on m-1 points plus the manipulations required for a cosine transform.

NEUMANN COUPLING PROCEDURE

To calculate (19b) (the coupling stage) a different approach is needed. Consider the transform

$$Y_i = \sum_{r=0}^{m-1} A_r \cos\left[\frac{(2i+1) r \pi}{2m}\right], \quad i = 0, 1, \ldots m-1 \quad (24)$$

and expand as

$$Y_i = \sum_{r=0}^{m-1} A_r \left(\cos\frac{i r \pi}{m} \cos\frac{r\pi}{2m} - \sin\frac{i r \pi}{m} \sin\frac{r\pi}{2m}\right).$$

This is identifiable with the sine-cosine series on 2m real points

$$Y_i = \tfrac{1}{2} a_0 + \sum_{r=1}^{m-1} a_r \cos\frac{i r \pi}{m} + \sum_{r=1}^{m-1} b_r \sin\frac{i r \pi}{m} + \tfrac{1}{2}(-1)^i a_m, \quad (25)$$

if

$$a_r = A_r \cos\frac{r\pi}{2m}, \quad r = 1, 2, \ldots m-1,$$
$$a_m = 0, \quad a_0 = 2A_0,$$

and

$$b_r = -A_r \sin\frac{r\pi}{2m}, \quad r = 1, 2, \ldots m-1.$$

Clearly the first m-values of Y_i in (25) are those required by (24). Again the transform problem has been embedded in a larger one of known type. It can easily be shown that the C_r's in

$$Y_i = \sum_{r=0}^{2M-1} C_r W_{2m}^{ri}, \quad i=0,1,\ldots 2M-1, \qquad (\bullet)$$

can be identified with the coefficients in (25) as follows:

$$a_r = 2\,\mathcal{R}(C_r), \quad 0 \leq r \leq m,$$

and

$$b_r = -2\,\mathcal{I}(C_r), \quad 0 < r < m.$$

The C_r's must of course be conjugate even, ie $C_r = \overline{C}_{2m-r}$.

The procedure for computing the sine-cosine transform (25) or (26) is set out by Cooley (1970). Using suitable formulae, the sequence C_r, $r=0,1,\ldots 2m-1$ is cast into another complex sequence A_r, $r=0,1,\ldots m-1$ and the transform

$$Z_i = \sum_{r=0}^{m-1} A_r W_m^{ri}, \quad i=0,1,\ldots m-1 \qquad (27)$$

computed. The result is a complex sequence Z_i whose real and imaginary parts are related to the Y_i defined in (25) by

$$Y(2i) = \mathcal{R}(Z_i),$$
$$Y(2i+1) = \mathcal{I}(Z_i), \quad i=0,1,\ldots m-1.$$

Hence the Y_i's required in (24) are the first $m/2$ (m even) complex values given by (27). Thus the coupling stage for the Neumann problem requires one half of a sine-cosine transform on 2m points.

§6. COMPARISON WITH ITERATIVE METHODS

In this section a comparison is made between the Fourier plus Double Sweep method discussed earlier and the efficient alterating direction implicit (ADI) iterative method of Peaceman and Rachford (1955). Since ADI is deemed one of the most efficient of iterative schemes, comparisons with other methods are not considered.

Since the methods can be compared in a variety of ways depending on the number and type of floating point operations, the aim here is to convert the operation

count per grid point into machine cycles required on an IBM 360/195 computer. For ADI an efficient scheme requires 6p multiplications and 10p additions per point, where p is the number of sweeps through the data. Counting two cycles per addition and three for a multiplication results in some 38p cycles per grid point for ADI.

Temperton (1976) gives the approximate number of floating point operations for Fourier transforms as

$$1.5p + 2.67q + 2.75r + 45 + \sum_i \tfrac{1}{2}(5+m_i)t_i - 1 + K_1,$$

additions, and

$$p + 2q + 1.5r + 3.25 + \sum_i \tfrac{1}{2}(2+m_i)t_i - 2 + K_2,$$

multiplications, where

$$n = 2^p . 3^q . 4^r . 5^s . m_1^{t_1} . m_2^{t_2} \ldots ,$$

and K_1, K_2 depend on the type of transform being performed. Of course Fourier methods are most efficient when n factors into small prime numbers - the efficiency depending greatly on the size of the largest factor.

To give an indication of the efficiencies that can be achieved a comparison is made on a grid having m=60 points in the x-direction.

DIRICHLET CONDITIONS

This is the easiest to consider and requires roughly 14 additions (K_1=5) and 8 multiplications (K_2=2.5) per grid point; ie about 42 cycles. The Gaussian step requires 25 cycles (including the division). Thus the total number of cycles required by the direct method would be 2x52 + 25 = 129 cycles, which is roughly equivalent to three-and-a-half ADI sweeps.

NEUMANN CONDITIONS

This is a little more tricky to compute, but as a guide it may be considered as one sine transform on m-1 points plus the manipulations for a cosine transform (decoupling stage) plus half of a sine-cosine transform on 2m real points (coupling stage). In this case in total there are roughly 38 multiplications, 22 additions and

1 division requiring about 150 cycles. Four ADI sweeps require 152 cycles. Because of the extra and varied manipulations required in the Neumann case, the true value may be higher than this.

In the above comparisons no account has been taken of the amount of time spent in actually indexing the arrays. This is an important secondary factor and should always be borne in mind when comparing methods under test conditions. Thus a scheme which halves the number of floating point operations may not halve the execution time if a different array indexing scheme has to be employed. Nevertheless, in general, any method which reduces substantially the floating point arithmetic is to be preferred since round-off error is also reduced.

The approach described in this note demonstrates that for the Helmholtz equation (1) a direct method of solution can compare very favourably with iterative methods.

REFERENCES

Bellman, R	1969	'Stability, Theory of Differential Equations'. Dover Pub., New York.
	1970	'Introduction to Matrix Analysis'. McGraw Hill.
Bickely, W.S. and McNamee, J.	1960	'Matrix and other direct methods for the solution of systems of linear difference equations'. Phil. Trans., 1005, 252, pp 69-131.
Cooley, J.W. Lewis, R.A.W. and Welch, P.D.	1967	'Histroical notes on the fast Fourier transform'. Proc. IEEE 55, p.1675 et seq.
	1970	'The fast Fourier transform algorithm: programming considerations in the calculation of sine, cosine and Laplace transforms'. J. Sound Vib., 12(3), pp. 315-337.
Osbourne, M.R.	1956	'Direct methods for the solution of finite-difference approximations to separable partial differential equations'. The Computer Journal, 8(20), pp. 155-156
Peaceman, D.W. and Rachford, H.H.	1955	'The numerical solution of parabolic and elliptic differential equations'. J. Soc. Indust. App. Maths., 3, pp. 28-41.
Richtmyer, R.D. and Morton, K.W.	1967	'Difference Methods for Initial Value Problems'. John Wiley
Temperton, C.	1976	'An all-purpose mixed-radix fast Fourier transform'. Met O 2b Tech. Note No 25.
Varga, R.S.	1962	'Matrix Iterative Analysis'. Prentice Hall.

ADDITIONAL NOTE ON THE NEUMANN COUPLING PROCEDURE.

When applying the methods described in this Technical Note to solving the Helmholtz Equations occurring in the mesoscale model, an improved method for handling the Neumann coupling procedure was found. Instead of transforming directly the m-vector represented by A_r $(r = 0, 1, \ldots m-1)$, consider the problem of computing the sine transform of A'_r, namely

$$Y'_i = \sum_{r=1}^{m-1} A'_r \sin \frac{\pi r i}{m}, \quad i = 1, 2, \ldots m-1, \tag{24a}$$

where

$$A'_r = -2 \sin\left(\frac{\pi r}{2m}\right) A_r .$$

From Equation (24) it is easy to show, using trigonometric relations, that

$$Y'_i = Y_i - Y_{i-1}, \quad i = 1, 2, \ldots m-1,$$

or

$$Y_i = Y_i' + Y_{i-1} . \tag{25a}$$

To complete the calculation of the Y_i's, one of them must be computed directly using equation (24), say Y_0. $Y_i \; (i = 1, \ldots m-1)$ may then be calculated from equation (25a). Hence the problem of computing the transform (24) may be reduced to that of a sine transform on m-1 points. Cooley (1970) gives the steps required to compute a basic sine transform.